CL50 Pro Process Data Function

11/9/2022

This document covers the installation and use of a function for Siemen's TIA Portal software package. This function handles cyclic IO-Link Process Data Out to a Banner CL50 Pro light via an IO-Link Master from a Siemens PLC. The function covers parsing and display of the CL50 Pro sensor Process Data Out.
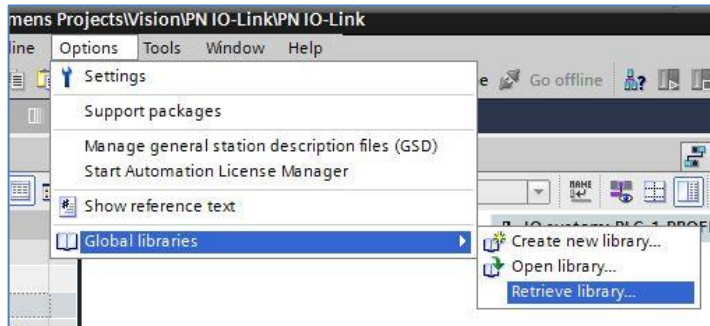
Components

Banner CL50 Library.zal14

There are two methods for the process data. The first is used when creating a connection to Banner's IO-Link masters. The second set of instructions are for systems using other manufacturer's IO-Link masters.

Installation Instructions

1. Open a project.
2. Go to Options > Global Libraries > Retrieve Library.

3. Select the Banner CL50 Library. Click Open.
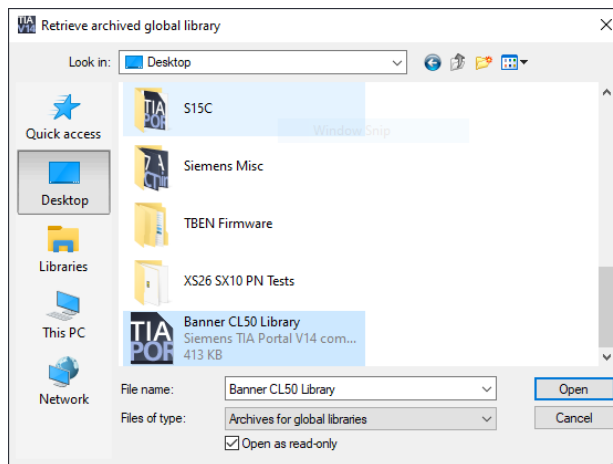
4. The library is now accessible in the Libraries tab.
5. Go to page 3 for Banner IO-Link Masters and to page 6 for all other IO-Link Masters.

Setup of CL50 with a Banner DXMR90-4K

1. Go to Device and Networks to configure the DXMR90-4K. Add the DXMR90-4K if it has yet to be added to the system.
2. Add Banner IO-Link Master Info to Slot 1. This sets the DXMR90-4K for IO-Link mode.
3. Open the IO-Link Generic Devices and select the proper module. The 4/4 byte is required for CL50. Make note of the I address for the Slot 2 which represents Port 1. Slot 2 starts at I1 for outputs. The other number needed is I3. The data for the port start at that point (I3). The previous two bytes are Port Control.

▼ dxm	0	0			1-port Device
▶ Interface	0	0 X1			dxm
Banner IO-Link Master Info_1	0	1	1...9		Banner IO-Link Master Info
IO-Link In/Out 4/ 4 Byte + Status_1	0	2	10...17	1...18	IO-Link In/Out 4/ 4 Byte + Status

4. Drag the necessary tag from IOLM_Control > Banner > Banner_Specialized_Data_Types. The tag used in this example is "Banner_4out". This tag represents the full raw process data along with port status information.
5. Drag the necessary files from the CL50 Folder.
   a. Move Banner_CL50_PDO and Banner_CL50_Raw_Data to the PLC Data Types area.
   b. Move Banner_CL50_PD to the Program Blocks area.
6. Go to PLC Tags. Create two tags. One tag is for the full data structure while the second creates a tag to represent the raw Process Data from the IO-Link Master. In this example, Tag table_1 was created, then the tag "CL50Pro IOLM1 01 PDO" was created using a Data Type of "Banner_4out". This naming convention calls out the type of device in question as well as the specific IO-Link Master and port number where the sensor is connected. A different IO-Link Master might be named IOLM2 or IOLM3, for instance, and other specific sensors may be connected to different port numbers. The "Q" address found in step 2 (%Q1) is tied to this new tag. The second is "CL50 IOLM1 01 outRaw" and uses the "Q" address found in step 2 (%Q3). This is the tag that will be used in the Function block.
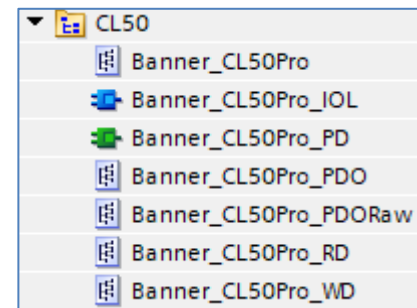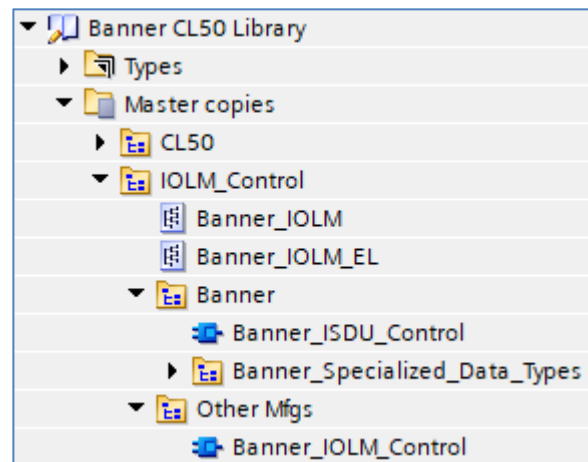
| Name	Data type	Address
▶ CL50 IOLM1 01 PDO	"Banner_4Out"	%Q1.0
▶ CL50 IOLM1 01 outRaw	"Banner_CL50Pro_PDORaw"	%Q3.0

7. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named "db".
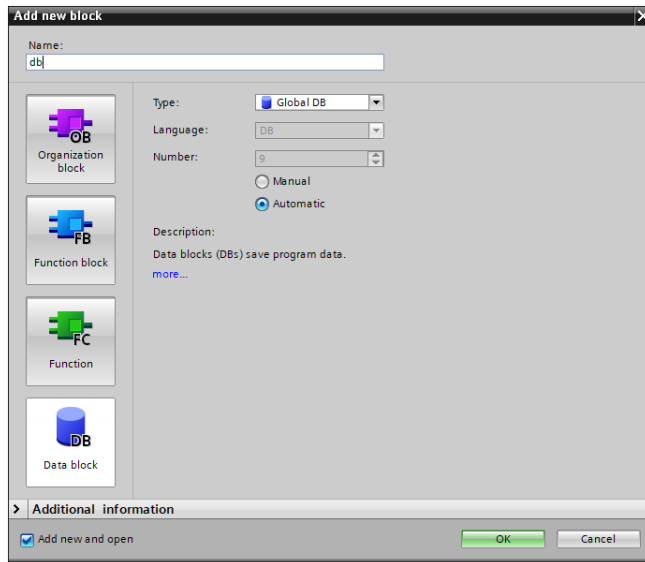

8. In the new data block, create a new tag to represent the parsed Process Data In for our Q4X. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type "Banner_CL50Pro_PDO" for the new tag.

CL50 IOLM1 01 PD	"Banner_CL50Pro_PDO"
Color 1	USInt
Color 2	USInt
Animation	USInt
Pulse Pattern	USInt
Speed	USInt
Color 1 Intensity	USInt
Color 2 Intensity	USInt
Audible State	USInt

8. Add the "Banner_CL50_PD" function to an OB lader. Link the "PDO" to the raw process data variable from step 5. The tag name again calls out the type of device, IO-Link Master, and the port number. Use the variable was called "CL50 IOLM1 01 outRaw" in this example. The "CL50 Pro PD" needs to be linked to the variable created in step 7. It was called "CL50 IOLM1 01 PD" for this example.
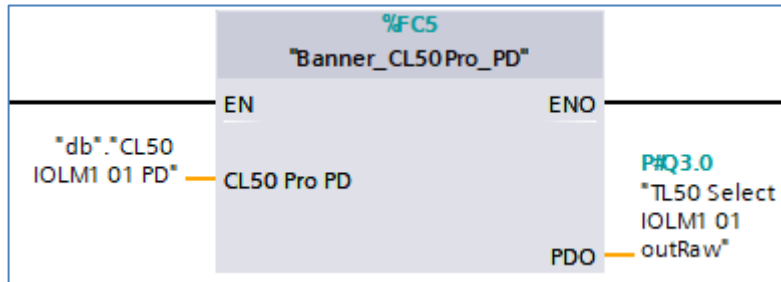

9. The final step is to configure the IO-Link output control. This is done by sending a 1 to Port Control and a 2 to Port Config. Both parameters are part of the tag created in step 6 "CL50 IOLM1 01 PDO".
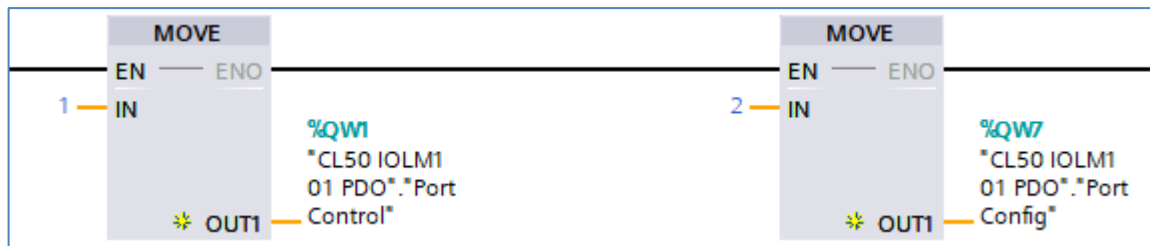
10. Process Data Setup is complete.
11. Compile and download the configuration to the PLC, then go online. Open the "db" data block and click Monitor all. The TL50 Select can be controlled now.

Setup of CL50 Select with other IO-Link Masters

1. The Banner CL50 Library will now be in the Global Library
    List. Expand the Master copies section. The CL50 folder
    contains elements for both Process Data and Parameter Data
    connections to a CL50 Pro device. As Process Data is the focus
    of this paper, we will concern ourselves with these items:
    Banner_CL50Pro_PD, Banner_CL50Pro_PDO, and
    Banner_CL50Pro_PDOraw.

2. Drag Banner_CL50Pro_PD to the Program Blocks area under
    your PLC.

3. Drag the other two items listed above to the PLC Data Types area under your PLC.

4. Go to Devices and networks to configure the system as necessary. Below is an example of
    what a configuration might look like. This example shows 5 different IO-Link Masters
   connected to the same PLC.
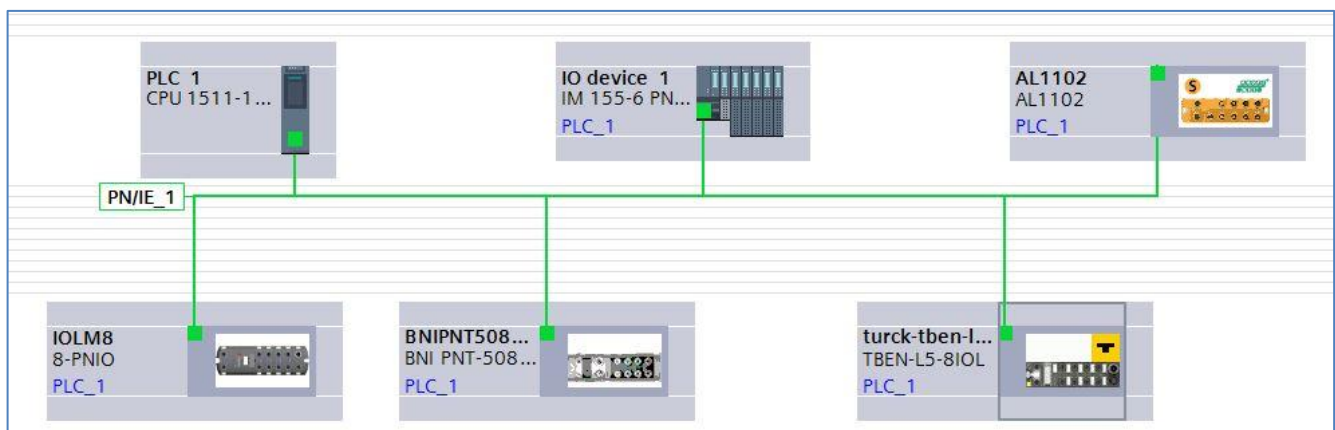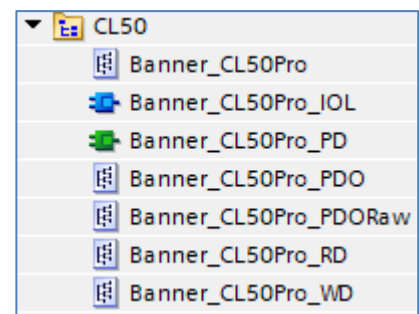
5. Click on the relevant device and configure the IO-Link Master as necessary. Refer to the
    documentation for the IO-Link Master. Recall that a CL50 Pro requires 3 bytes of space for the
    Process Data.

6. Record the "Q" address where this CL50 Pro Process Data is to be stored, as the address will be required in the next step. In this example, 3 bytes of Process Data Out for port 1 on the IO-Link Master will be stored in Q68 through Q70.

7. Go to PLC Tags. Add a new tag table, then create a new tag to represent the raw Process Data Out to be sent to the IO-Link Master. In this example, Tag table_1 was created, then the tag "CL50 IOLM1 01 PDO" was created using a Data Type of "Banner_CL50Pro_PDORaw". This naming convention calls out the type of sensor in question as well as the specific IO-Link Master and port number where the sensor is connected. A different IO-Link Master might be named IOLM2 or IOLM3, for instance, and other specific sensors may be connected to different port numbers. The "Q" address found in step 6 is tied to this new tag.

| Name	Data type	Address
▶ CL50 IOLM1 01 PDO	"Banner_CL50Pro_PDORaw"	%Q68.0

8. Go to Program blocks. Add a new Data block if necessary. In this example the new data block is named "Banner IO-Link Data".
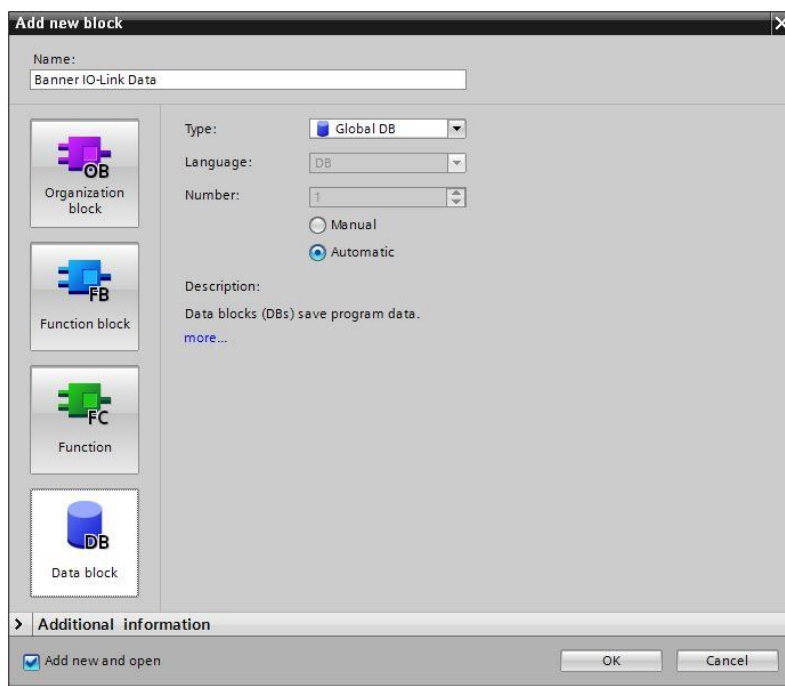
9. In the new data block, create a new tag to represent the parsed Process Data In for our CL50 Pro. The tag name again calls out the type of sensor, the IO-Link Master, and the port number. Use the data type "Banner_CL50Pro_PDO" for the new tag.

Banner IO-Link Data	
Name	Data type
1 ▼ Static	
2 ■ ▼ CL50 IOLM1 01 PD	"Banner_CL50Pro_PDO"

Add the "Banner_CL50Pro_PDO" function to an OB ladder. Link the "PDO" to the raw Process Data variable from step 7. Link "CL50 Pro PD" to the parsed Process Data variable from step 9.
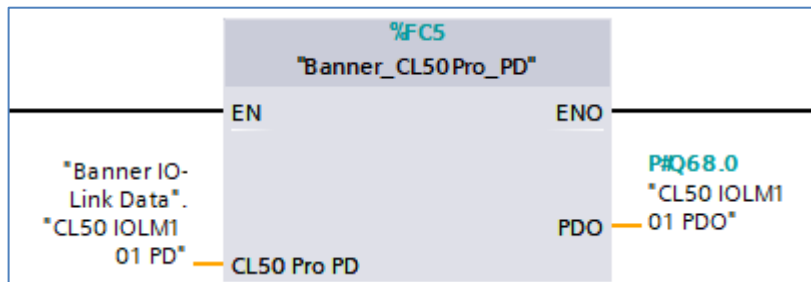
10. Process Data setup is complete.

11. Compile and download the configuration to the PLC, then go online. Open the "Banner IO-Link Data" data block and click Monitor all. You should see parsed CL50 Pro Process Data Out, like that shown below.

8

Banner IO-Link Data

		Name	Data type	Start value	Monitor value
1		▼ Static			
2		▼ CL50 IOLM1 01 PD	"Banner_CL50Pro_PDO"		
3		Color 1	USInt	0	3
4		Color 2	USInt	0	0
5		Animation	USInt	0	1
6		Pulse Pattern	USInt	0	0
7		Speed	USInt	0	0
8		Color 1 Intensity	USInt	0	0
9		Color 2 Intensity	USInt	0	0
10		Audible State	USInt	0	0

Appendix A   CL50 Pro Select Process Data Out

The CL50 Pro has 3 bytes of Process Data Out.

This Process Data is mapped to a specific group of PROFINET addresses. The 24-bits of Process Data encode many separate pieces of information, as shown below.

This function intelligently parses this Process Data into its component pieces.

ProcessDataOut "Process Data Out Run Mode" id=V_Pd_OutRunMode

bit length: 24
data type: 24-bit Record

subindex	bit offset	data type	allowed values	default value	acc. restr.	mod. other var.	excl. from DS	name	description
1	0	4-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Spring Green, 7 = Cyan, 8 = Sky Blue, 9 = Blue, 10 = Violet, 11 = Magenta, 12 = Rose, 13 = White, 14 = Custom 1, 15 = Custom 2					Color 1	The main color of the Animation. Custom Colors are defined in Parameter data
2	4	4-bit UInteger	0 = Green, 1 = Red, 2 = Orange, 3 = Amber, 4 = Yellow, 5 = Lime Green, 6 = Spring Green, 7 = Cyan, 8 = Sky Blue, 9 = Blue, 10 = Violet, 11 = Magenta, 12 = Rose, 13 = White, 14 = Custom 1, 15 = Custom 2					Color 2	The secondary color of the Animation. Only used if Animation has two colors. Custom Colors are defined in Parameter data
3	8	3-bit UInteger	0 = Off, 1 = Steady, 2 = Flash, 3 = Two Color Flash, 5 = Intensity Sweep					Animation	The Animation type
4	11	3-bit UInteger	0 = Normal, 1 = Strobe, 2 = Three Pulse, 3 = SOS, 4 = Random					Pulse Pattern	The pattern of Animation
5	14	2-bit UInteger	0 = Medium, 1 = Fast, 2 = Slow					Speed	The speed of the Animation
6	16	3-bit UInteger	0 = High, 1 = Low, 2 = Medium, 3 = Off, 4 = Custom					Color 1 Intensity	The Intensity of Color 1, Custom Intensity defined in Parameter Data
7	19	3-bit UInteger	0 = High, 1 = Low, 2 = Medium, 3 = Off, 4 = Custom					Color 2 Intensity	The Intensity of Color 2, Custom Intensity defined in Parameter Data
8	22	2-bit UInteger	0 = Off, 1 = On, 2 = Pulsed, 3 = SOS Pulse					Audible State	The state of the audible segment